

IP-Based design reuse for analog systems

Timothée Levi, Jean Tomas, Noëlle Lewis, Pascal Fouillat
IMS Laboratory, University Bordeaux 1, 351 Cours de la Libération, 33405 Talence, FRANCE;

ABSTRACT

The design flow of Analog and Mixed Signal has to be improved. In a specific application, we propose a definition of the IP content and the structure of an IP-based library. The case study consists in the neuron-level integration of a complete system that emulates spiking neural networks. As it is often the case, the development of the analog part of the system requires the largest amount of time, due to the lack of formalism and automation in that domain. One solution to accelerate the analog design cycle is to re-use already designed blocks and accumulated design knowledge, which could be illustrated by the IP (Intellectual Property) concept. Indeed, an experience of about ten years and 19 designed ASICs allow now to have an accurate idea of the system hierarchy and the recurrent analog blocks, which is the basis of IP-based design. We will describe the IP-based library which has been developed for that specific application domain and show how it can be used to accelerate the design cycle of the next ASIC generation.

Keywords: analog IP, design reuse, IP-based library

1. INTRODUCTION

The main objective of this work is to improve the design flow of analog and mixed signal integrated circuits. Purely digital systems profit from standardized tools and methods which allow automatic synthesis. In the analog domain, automatic synthesis has probably no sense and efforts must be done towards developing consistent and coherent CAD (Computer Aided Design) methods and tools.

The main idea is to re-use the accumulated design knowledge which could be illustrated by the IP (Intellectual Property) concept. In this paper, we give our experience on the following aspects: content of an analog IP, IP-based library and its integration in the system design flow, applied on a specific example.

This paper presents the neuron-level integration of a complete system that emulates spiking neural networks. Analog ASICs are used for the computation of neurons activity while the connectivity between these neurons is digitally controlled. One ASIC integrates several biologically realistic neurons, following the Hodgkin-Huxley formalism, where mathematical functions are emulated by typical or specific analog blocks.

2. APPLICATION DOMAIN

2.1 Neuromorphic engineering

Engineering of neuromorphic integrated systems is a research field where microelectronics encounters biology. The link between both is realized by computational neurosciences which model and emulate a part of brain activity. Different levels of modelling exist from the neuron physiology to the plasticity of large neurons networks. One issue is to have the adequate simulation system that implements those models; that is the role of neuromorphic engineering [1], [2]. Important features of such systems are re-configurability, observability and also real-time running, especially for hybrid (real/artificial) experiments.

From the microelectronic point of view, one solution is to design analog ASICs for the real-time computation of neurons activity and to digitally control the connectivity between these neurons. As it is often the case, the development of the analog part requires the largest amount of time, due to the lack of formalism and automation in that domain.

2.2 Case study: neuromimetic analog ASIC

To design neuromimetic IC (in contrast to bio-inspired IC), we chose the Hodgkin-Huxley formalism. The main advantage of this formalism is that it relies on parameters, which are biophysically realistic, by the way of a conductance-based expression of the neural activity (see detailed description of the Hodgkin-Huxley formalism in [3]). We will express here the conductance-based principle and find out generic expressions for these conductance phenomena.

The Hodgkin-Huxley formalism provides a set of equations and an electrical equivalent circuit (Fig. 1) that describe the behavior of separate conductances. Each conductance represents the dynamics of an ionic specie (sodium, potassium or calcium) flowing through the neural membrane. All these ionic currents are integrated on the membrane capacitance following the electrical equation (1),

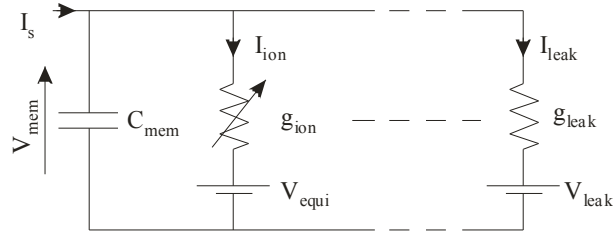


Figure 1: Neuron electrical equivalent circuit

$$C_{mem} \frac{dV_{mem}}{dt} = -\sum I_{ion} + I_s \quad (1)$$

where V_{mem} is the membrane potential, C_{mem} the membrane capacitance and I_s is an eventual stimulation or synaptic current. The generic mathematical expression (2) for each ionic current is:

$$I_{ion} = g_{max} m^p h^q (V_{mem} - V_{equi}) \quad (2)$$

$$m_{\infty}(V_{mem}) = \frac{1}{1 + \exp(\pm \frac{V_{mem} - V_{offset}}{V_{slope}})} \quad (3)$$

$$\tau(V_{mem}) \cdot \frac{dm}{dt} = m_{\infty}(V_{mem}) - m \quad (4)$$

where g_{max} is the maximal conductance value, m and h are respectively the activation and inactivation function, V_{equi} the ion-specific reverse potential, and τ is the kinetic.. For our implementation we will integrate current generators that follow expression (2). Synaptic current are based on similar conductance-based model.

All model equations can be implemented using classical analog building blocks like differential amplifiers, OTAs, current mirrors and current-mode multipliers. For example OTA are used to implement the transconductance (g_{max}). The parameters m and h are described by equations (3) and (4) and require one sigmoidal circuit and kinetic blocks. Neuromimetic ASICs have been designed by our team [4] from 1993. Now, the trend is to integrate more and more complex neural networks, on the basis on recurrent primitive analog blocks. At the beginning of the design process, the specifications are given by neuroscientists and biologists; they consist mainly in parameters controlling the *block level* but also in some data related to the network topology.

Due to the increasing complexity of networks to be implemented, it becomes crucial to maximize the re-using of previously designed blocks.

The following paragraphs will describe firstly what could be embedded in an analog IP, then how these entities may be organised in a data-base and finally how to use this data-base in the ASIC design flow.

3. DEFINITION OF AN ANALOG IP

3.1 Analog IP content

Defining the IP content is the first important task because the IP concept is the base of design re-use. Some works already exist towards the re-use of AMS blocks [5] [6] [7].

What should be the main properties of an IP block ?

First, it should give a precise characterization of already designed block; second, it has to be described with adequate representations or models, consistent with the design levels, to enable the easy re-use of the block, along the complete design flow.

Table 1 shows the different descriptions (or *views*) that are embedded in our IP blocks.

All these views have the same terminals and the same symbol.

Table 1: IP-AMS hierarchical description levels

View name	Description / Role
Symbol	visualizes the function
Connectical	verifies connexion between blocks (VerilogA)
Functional	models ideal electrical behaviour (VerilogA)
Behavioural	models non-ideal electrical behaviour, extracted from schematic (VerilogA)
Schematic	transistor-level schematic
Layout	
Characterization	See below ...

For the *connectical*, *functional* and *behavioural* views, we use verilogA language. These views are useful for multilevel simulations especially in the verification phase of the design process. The *functional* view describes the ideal equations of the function to be implemented. The *behavioural* view is more detailed indeed there are refined equations which fit the schematic behaviour.

One important point is to have a fluent and coherent design flow, that is why logical and mathematical links have been established between the different views.

The *characterization* view is the most important view for the research of the corresponding IP. This view is the main point of the re-use methodology. It contains information about the design and the re-use ability of the IP-block. The technology, the supply voltage, the terminals and their validity domains, the links between the functional model and the behavioural model, the area of the layout and the cost of the design (which may be calculated with the technology and the area) are defined in this file.

The validity domain gives the range of input signals, where the circuit functionality is conserved. This information comes from systematic simulation of the primitive *cells* (*cell level*); then it is propagated to the *blocks* and *macro-blocks*.

3.2 Example of IP: kinetic cell

The kinetic function is one of the basic functions implemented in our ASIC. We give here just the main views of the bloc. All the graphic views and the behavioural views are not shown in this paper.

- **Biological equation to be implemented :**

$$x_{activation} = x_{inf\ ini} - Tau \cdot \frac{dx_{activation}}{dt} \quad (5)$$

- **Transformation to the electrical functional model:**

Some conversion factors are added between the biological model and the electrical one. Indeed for the current, the voltage and the conductance we have to assure better signal dynamic performance for the robustness of the circuit. The two models use the same timing scale.

In the electrical functional model, $x_{activation}$ is represented by a current, so the model equation becomes:

$$I_{activation} = I_{inf\ ini} - Tau \cdot \frac{dI_{activation}}{dt} \quad (6)$$

- **Characterization of the sigmoid cell:**

This component has five terminals.

Table 2: IP-AMS hierarchical description levels

Technology	BiCMOS AMS 0.35 μm
Supply voltage	5 V
Inputs	Iin, Vcoef, Capa, Ih_10u
Outputs	Idiff
Layout	20990 μm^2

- **Validity domains:**

The Table 3 describes the validity domains of some inputs of the kinetic IP.

Table 3: IP-AMS hierarchical description levels

Inputs	Minimum	Maximum
Iin	0 μA	20 μA
Vcoef	-300 mV	400 mV

- Relation between the functional and the behavioural model:

For the behavioural model, the detailed analysis of the schematic gives a refined equation:

$$Tau = \frac{I_{h_10u} \cdot R_{conv} \cdot C}{(V_{cc} - V_{coef}) \cdot g_{OTA}} \quad (7)$$

4. IP-BASED LIBRARY

4.1 Characteristics of the data-base

The data-base must implement:

- The hierarchy of the system
- The IPs and their content

For our application, the hierarchy is well established.

The data-base is implemented with MySQL [8] formalism. Nine tables have been created:

- four tables are for the link father-son between the different entities of the hierarchy
- one table defines each IP
- one table gives the number of IPs available for each function
- three other tables defines the set of model parameters for each hierarchical level

4.2 Data-base exploration

The objective is to find one ASIC solution according to the initial specifications.

The chosen method is to perform a Top-Down exploration, from the *macro-block level* to the *cell level* (figure 2) using the validity domain as a selection criterium. If we have different corresponding IPs, we should choose the one with the largest validity's domain.

The exploration requests are made in Php [9]. These requests are created from the system specifications.

At the final step, a diagnostic is returned which quantifies the possibility to re-use IPs for a new ASIC project. In a case of re-use, IPs netlists are created for the designer.

4.3 Results of the exploration

After the Top-Down exploration, the designer has the summary of the exploration. If a system is found, it has the complete list of IPs (figure 5) with their location and description view, in an html format. Then he could use the created netlist to verify the system with the behavioural views. If a system is not found, a listing of errors is done and one help of reuse is given.

5. APPLICATION EXAMPLE

5.1 Example description

As illustrated on Fig. 2, the ASIC is composed of neurons which are defined by some ionic currents which are made by some elementary functions.

The ASIC represents the *system level*, a neuron the *macro-block level*, an ionic current the *block level* and an elementary function the *cell level*.

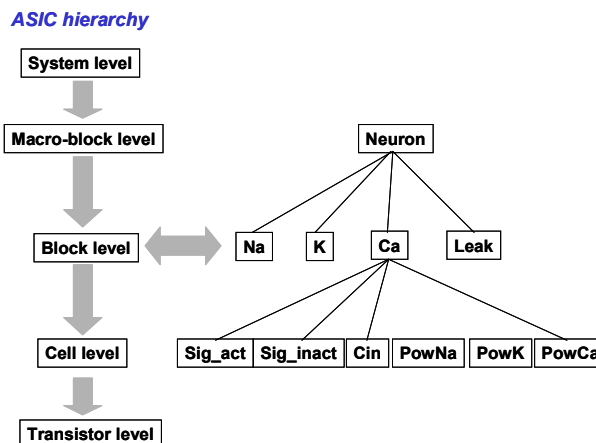


Figure 2: Hierarchy of the ASIC

Each cell contains some IPs which are defined previously.

5.2 Specifications

The system specifications are given by the neurobiologists (figure 3). Then these specifications are converted in electrical ones. Finally, requests are created by these electrical specifications.

Figure 5 is a shoot screen of the software’s interface which allows biologists to enter the specifications.

The parameters corresponding to the system specifications are summarized in Table 4.

Summary of the system specifications of Levi

You already have an account

You want to have 2 neurons in your network

Neuron 1

Cmem_bio in uF/cm2

Area of neuron 1 in cm2

Please choose your ionic currents

Sodium
 Potassium
 Leak
 Calcium
 5th ionic current

How many multihyapses for this neuron do you want ?

Neuron 2

Cmem_bio in uF/cm2

Area of neuron 2 in cm2

Please choose your ionic currents

Sodium
 Potassium
 Leak
 Calcium
 5th ionic current

How many multihyapses for this neuron do you want ?

[Come back to the initial system specifications](#)

Figure 3: System specifications

Table 4: Parameters of the system specifications

Currents	Voffset (mV)	Vslope (mV)	Tau (ms)	Vequi (mV)	gmax (S)
Neuron 1					
Na	-37	7.2	0.03	50	110 μ
K	-37	11.38	3	-100	22 μ
Ca	-35	11.4	8	-700	50 μ
Leak					220n
Neuron 2					
Na	-41	5.7	0.08	80	70 μ
K	-31	13.81	5	-120	13 μ
Leak					310n

A Top-Down exploration is performed, from the *macro-block level* to the *cell level* using the validity domain as a selection criterium. If we have different corresponding IPs, we should choose the one with the largest validity's domain.

The exploration requests are made in Php. These requests are created from the system specifications.

At the final step, a diagnostic on re-usable blocks is returned which quantifies the possibility to re-use IPs for a new ASIC project. In a case of re-use, IPs hierarchical netlists are automatically created for the designer.

5.3 Results

The data-base automatic exploration process has been tested for many plausible specifications set. An example of a diagnostic is given in Table 3 and Table 4. It describes the IPs we can re-use.

Table 3: Validation of the ionic currents

Ionic currents	Neuron 1	Neuron 2
Na	Right	Right
K	Right	Right
Ca	Wrong	
Leak	Right	Right

Table 4: Final diagnostic

Neurons	Numbers of IPs we can re-use	Percentage of re-use for this circuit
Neuron 1	60/68	88 %
Neuron 2	60/60	100 %

Table 3 describes an error in the re-use of the Ca ionic current. Indeed one parameter of the specifications ($V_{leak} = -700$ mV) is not included in each validity domain of Ca IPs. That's why we cannot find an IP for these specifications. So the designer has to build another IP which fits with the parameters.

**** Neuron recovery **** : ip1_block_system

Level	0
values	0
symbol	paminalblock_systemsymbolsymbol.cdb
connectical	paminalblock_systemconnectical.txt
functional_model	paminalblock_systemfunctional.txt
behavioural_model	paminalblock_systembehavioural.txt
schematic	paminalblock_systemschematic.sch.cdb
layout	paminalblock_systemlayoutlayout.cdb
technology	AMS BiCMOS 0.35 um
supply_voltage	5
Nb_inputs	1
Nb_outputs	19
layout_area	0
ip_cost	0
datasheet	ip1_block_system_datasheet.pdf

[Return to the Top](#)

[Return to the last Neuron](#)

**** Neuron recovery **** : ip1_neuron_rs_final

Figure 5: snapshot of the html result file

5.4 Verification

The Bottom-Up phase verifies if the proposed architecture really meets the specifications. Indeed the behavioral and functional models are used to perform a multi-level simulation of the entire ASIC.

All the blocks have been validated using the analog simulator *Spectre* under *Cadence* environment. Due to the huge number of components that compose the chip, it is not possible to perform transistor-level simulation of the final chip. Furthermore, the neural activity is a low frequency activity, and the neural activity has to be simulated for at least tens of ms as shown in figure 6.

We decided as a consequence to mix transistor level, behavioral, functional and connectical descriptions of the different blocks to validate their connectivity and ensure the functionality of the whole design.

As an example, a simulation for one neuron of type FS (Fast Spiking [10]) with null synapses inputs lasts 2'17" using *VerilogA* description and 14'56" using transistor level description. Both simulations give the same neural activity as depicted in figure 6.

To perform the validation of the whole circuit, we used the *Hierarchical Editor* of *Cadence* environment. For each simulation, only one block was described at the transistor level while the others were described using behavioral description.

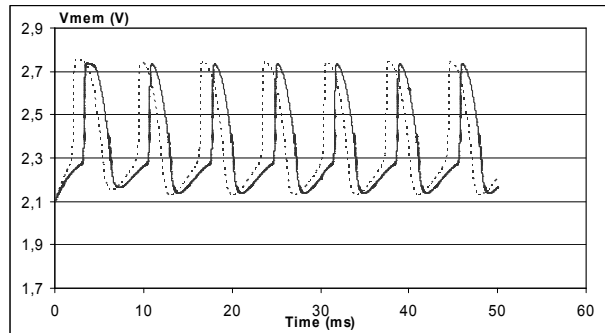


Figure 6: electrical activity of a FS spiking neuron (dashed line corresponds to VerilogA description and continuous line to transistor level description).

6. CONCLUSION

In this paper, a case-study in analog IP re-use is exposed. The questions that have been developed are: analog IP definition, IP-based library and IP-based system design. To accelerate the time of design of our system and improve the design re-use, we define an IP-based library of analog elementary functions and a specific database of already designed blocks. This database allows us to be able to build more rapidly the next generations of ASICs (Top-Down Phase), with varying numbers of ionic and synaptic conductances. The library associates behavioral and schematic views of all blocks at all hierarchical level, to be able to perform *Spectre* simulations of the whole chip (Bottom-Up Phase) in a reasonable CPU time. Simulation time is a key issue in our case, considering the complexity of the Kirchoff network of the circuit. This exploration allows obtaining a rapid estimation of the amount of re-usable circuits.

REFERENCES

1. S. Saïghi, Y. Bornat, J. Tomas, S. Renaud, "Neuromimetic ICs and System for Parameters Extraction in Biological Neuron Models", Proceedings ISCAS 2006, pp.4207-4210, Island of Kos, Greece, May 2006
2. Y. Bornat, J. Tomas, S. Saïghi, S. Renaud, "BiCMOS Analog Integrated Circuits for Embedded Spiking Neural Networks", Proceedings DCIS 2005, Lisbon, Portugal, November 2005
3. A.L. Hodgkin, A.F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve", Journal of Physiology, 1952
4. <http://neuromorphic.ixl.fr>
5. Z. Li, L. Luo, J. Yuan, "A Study on Analog IP Blocks for Mixed-Signal SoC", Proceedings ASIC, pp.564-567, Beijing, China, October 2003
6. I. O'Connor, F. Tissafi-Drissi, G. Révy, F. Gaffiot, "UML/XML-based approach to hierarchical AMS synthesis", Proceedings FDL 2005, Lausanne, Switzerland, September 2005
7. N. Martínez Madrid, E. Peralías, A. Acosta, A. Rueda, "Analog/Mixed-Signal IP modelling for design reuse", DATE Conference, Munich, Germany, 2001
8. P. Dubois, MySQL, Sams Developer's Library, 2005
9. R. Lerdorf, K. Tatroe, Programming Php, O'Reilly publisher, 2002
10. B. Connors, M. Gutnick, "Intrinsic firing patterns of diverse neocortical neurons", Trends Neuroscience, vol.13, pp. 99-104, 1990